File permissions in Linux

Project description

In this project, I acted as a security professional managing file permissions for a research team using Linux. I reviewed existing permissions to ensure they matched authorization policies and updated them using Linux commands. My goal was to prevent unauthorized access by properly configuring permissions on files and directories. Through this, I practiced key commands like `ls -la` and `chmod`.

Check file and directory details

```
ls -la
```

This lists all files and directories, including hidden ones, with their permissions, owner, group, size, and date.

Describe the permissions string

```
-rw-r--r--
```

This means:
- `-`: regular file
- `rw-`: owner can read & write
- `r--`: group can read
- `r--`: others can read

Only the owner can change this file.

Change file permissions

```
chmod 640 report.txt
```

This allows:
- Owner: read + write
- Group: read only
- Others: no access

Change file permissions on a hidden file

```
chmod 600 .secret_notes.txt
```

This gives full access to the owner only. Hidden files (start with `.`) usually store config or sensitive info.

Change directory permissions

```
chmod 750 project_folder/
```

This allows:
- Owner: read, write, execute
- Group: read, execute
- Others: no access

Useful for private team folders.

Summary

I used Linux to view and manage file permissions using `ls -la` and `chmod`. I secured files by adjusting access levels for owner, group, and others. I also worked with hidden files and directories to prevent unauthorized access. These skills are essential for protecting data in real-world environments.